Intel® Itanium® Processor Family System Abstraction Layer Specification Update, August 2005

Notice: Intel® Itanium® architecture processors may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are documented in processor specification updates.

Document Number: 309140-001

Contents

§

Revision History

Date	Version	Description
August 2005	001	Added PCI Express* Error Record; noted documentation changes in several sections.

§

Preface

This document is a compilation of documentation errata, specification clarifications, and changes applicable to the specifications contained in Table 1. Changes listed in this document are meant to supersede previous versions and may also contain previously unpublished information.

This document is intended for hardware system manufacturers and firmware/software developers.

Table 1. Affected Documents/Related Documents

Title	Document #
December 2003 release of the *Intel® Itanium® Processor Family System Abstraction Layer Specification*	245359-007
June 2004 release of the *Intel® Itanium® Processor Family System Abstraction Layer Specification* Update	252489-002

Nomenclature

Specification Changes are modifications to the current published specifications in the Affected Documents list and should be considered a part of those documents. These changes will be incorporated in the next release of the specifications.

Specification Clarifications describe a specification in greater detail or further explain a specification's interpretation. These clarifications will be incorporated in the next release of the specification.

Documentation Changes include typos, errors, or omissions from the current published specifications. These changes will be incorporated in the next release of the documents listed above.

§

Summary of Changes

The following tables indicate the specification changes, specification clarifications, or documentation changes that apply to the December 2003 release of the *Intel® Itanium® Processor Family System Abstraction Layer Specification*.

Specification Changes

No.	Page	SPECIFICATION CHANGES
1	8	PCIe* Error Record Section

Specification Clarifications

No.	Page	SPECIFICATION CLARIFICATIONS
		None for this revision of the Specification Update

Documentation Changes

No.	Page	DOCUMENTATION CHANGES
1	12	Clarification to Platform PCI Component Error Info Error Record Section
	12	
	12	
4	12	SAL Procedure Return Value

§

Specification Changes

1. PCIe* Error Record Section

A new Error Record section for PCI Express* devices has been created to work with PCIe Advanced Error Reporting. Add the following section to Appendix B - Error Record Structures:

B.2.4.7 Platform PCIe* Error Info

This section describes the errors that occur on the PCI Express, for Root Port (RP), PCIe Bridge, Switch and End Point (EP) device. The error information for the PCIe RP is as reported by the RP bridge. An error within a PCIe component that results in error signalling on the PCIe bus will result in multiple sections being reported, one or more sections for each of the RP and one or more sections for the downstream PCIe devices.

In a system configured to enable AER, but with the presence of a downstream PCI/PCI-X bridge and its associated endpoint devices, the error record returned may consist of PCIe error sections with PCIe device types of 1, 7or 8. However, system firmware can optionally provide the PCI Bus and PCI Component error sections alongside PCIe error sections in a single error record.

The ownership of the AER resources is negotiated between SAL and OS through an ACPI interface defined in the PCI Firmware Specification version 3.0. SAL must not control or report on PCIe error resources, if the OS has taken native control of AER. SAL is allowed to provide the PCIe error record sections to the OS when SAL owns the AER resources. SAL owns the PCIe error resources on systems in that boot an OS which is not ACPI 3.0 compliant or PCIe aware.

Offset	Length	Field	Description
0	16 bytes	GUID	{ 0x98d1e922, 0xd358, 0x4d22, { 0x98, 0xf3, 0xac, 0xad, 0xdc, 0x4c, 0x3b, 0x6 } }
16-23	8 bytes		See Section B.2.2 for details.

PLATFORM_PCIe_BUS_ERROR_RECORD SECTION BODY STRUCTURE

Offset	Length	Field	Description
0	8	VALIDATION_BITS	Validation Bits to indicate the validity of each of the subsequent fields: Bit 0- PCIe_DEVICE_PORT_TYPE_VALID_BIT Bit 1– PCIe_VERSION_INFO_VALID_BIT Bit 2– PCIe_CMD_STS_INFO_VALID_BIT Bit 3– PCIe_DEVICE_ID_INFO_VALID_BIT Bit 4– PCIe_DEVICE_SN_INFO_VALID_BIT Bit 5– PCIe_BRIDGE_CNTRL_STS_INFO_VALID_BIT Bit 6– PCIE_CAP_STRUCT_CNTRL_STS_INFO_VALID_BIT Bit 7– PCIe_AER_INFO_STRUCT_VALID_BIT Bit 8-15: Reserved Bit 16– PCIe_BUS_OEM_ID_VALID_BIT Bit 17 – PCIe_BUS_OEM_DATA_STRUCT_VALID_BIT Bit 18..63– RESERVED
8	4 bytes	PCIe_DEVICE_PORT_TYPE	PCIe Device/Port Type as defined in the PCI Express capabilities register[a]. Refer to PCIe specification for more details on the following encoding. The encoding per the PCIe specification is as follows: 0: PCI Express End Point 1: Legacy PCI End Point Device 4: Root Port of root complex 5: Upstream port for PCI Express Switch 6: Downstream port for PCI Express Switch 7: PCI Express to PCI/PCI-X Bridge 8: PCI/PCI-X to PCI Express bridge 9. Root Complex Integrated Endpoint Device 10. Root Complex Event Collector
12	4 bytes	PCIe_VERSION_INFO	PCIe Spec. version number supported by the platform Byte0-1: PCIe Spec. Version Number • Byte0: Minor Version in BCD • Btye1: Major Version in BCD Byte2-3: Reserved
16	4 bytes	PCIe_CMD_STS_INFO	PCIe Root Port Bridge or Device PCI compatible Command & Status Register Information. Byte0-1: PCI Command Register Byte2-3: PCI Status Register
20	4 bytes	Reserved	Reserved

Offset	Length	Field	Description
24	16 bytes	PCIe_DEVICE_ID_INFO	PCIe Root Port PCI/bridge PCI compatible device number and bus number information to uniquely identify the a root port or bridge. Default values for both the bus numbers is zero Byte 0-1: Vendor ID Byte 2-3: Device ID Byte 4-6: Class Code Byte7: Function Number Byte8: Device Number Byte9-10: Segment Number Byte11: Root Port/Bridge Primary Bus Number or device bus number Byte12: Root Port/Bridge Secondary Bus Number Byte13-14: bit0-2: Reserved. bit3:15: Slot number as defined in the PCIe Slot Capability register for "Physical Slot Number" field Byte 15: Reserved
40	8 bytes	PCIe_DEVICE_SN_INFO	PCIe Root Port PCI/bridge or end point device serial number Byte0-3: PCIe Device Serial Number Lower DW Byte4-7: PCIe Device Serial Number Upper DW
48	4 bytes	PCIe_BRIDGE_CNTRL_STS_INFO	PCIe Root Port/Bridge Secondary Status & Control Register Information. This field is valid for bridges only. Byte0-1: Bridge Secondary Status Register Byte2-3: Bridge Control Register
52	36 bytes	PCIe_CAP_STRUCT_CNTRL_STS_INFO	PCIe Capability Structure The 36-byte structure containing device capabilities and status, as defined in the PCIe[b] Base Specification. The fields in the structure vary with different device types. The “Next CAP pointer” field should be considered invalid and any reserved fields of the structure are reserved for future use. Note that PCIe devices without AER (PCIe_AER_INFO_STRUCT_VALID_BIT=0) may report status using this structure!
88	96 bytes	PCIe_AER_INFO_STRUCT	PCIe Advance Error Reporting Extended Capability Structure corresponding to the device type specified in the capability register or the header. Refer to PCIe[c] and PCIe to PCI/PCI-X bridge[d] specification for details on the AER information for Root Port, End Point Device, PCIe bridge, PCI Switch and etc. The fields in the structure will vary with different device types. The unused part of the structure is reserved for future use. For devices without AER capabilities this field is invalid and PCIe_AER_INFO_STRUCT_VALID_BIT should be set to 0.
184	16 bytes	PCIe_BUS_OEM_ID	OEM specific data containing identification information for the PCIe Bus.
200	N bytes	PCIe_BUS_OEM_DATA_STRUCT	OEM specific data of variable length. See Table B-4 for the format of this structure. N equals 0 if the PCIe_BUS_OEM_DATA_STRUCT_VALID_BIT is not set.

a.Refer to Section 7.8.2, Table 7-10 of PCI Express Base Spec., Rev. 1.1
b.Refer to Section 7.8, Figure 7-9 of PCI Express Base Spec., Rev. 1.1
c.Refer to Section 7.10, Figure 7-26 of PCI Express Base Spec., Rev. 1.1
d.Refer to Section 5.2.3, Figure 5-4 of PCI Express to PCI/PCI-X Bridge Spec., Rev. 1.0

Refer to the *PCIe* Specification (http://www.pcisig.com)* for further details.

§

Specification Clarifications

There are no Specification Clarifications for this revision of the *Intel® Itanium® Processor Family System Abstraction Layer Specification Update, August 2005*.

§

Documentation Changes

1. Clarification to Platform PCI Component Error Info Error Record Section

In the Platform PCI Component Error Info Error Record Section of B.2.4.3, replace the PCI_COMP_REGS_DATA_PAIR and PCI_COM_OEM_DATA_STRUCT fields with the following clarification:

40	2 x 8 x M bytes	PCI_COMP_REGS_ DATA_PAIR	An array of address/data pair values. The data may be 8 bytes in length. M = PCI_COMP_MEM_NUM + PCI_COMP_IO_NUM
40+2 x8x M	N bytes	PCI_COMP_OEM_ DATA_STRUCT	OEM specific data of variable length. See Table B-2 for the format of this structure. If PCI_COMP_OEM_DATA_STRUCT_VALID_BIT is not set, N = 0.

2. Error Record Header Minor Revision Number Increment

The addition of the PCIe error record section requires a new minor revision number in the Error Record Header. The Revision field of the Record Header in Section B.2.1 must be replaced with the following:

8	2 bytes	REVISION	2-byte Major and Minor revision number of the Record in BCD format: Byte0 – Minor (0x05) Byte1 – Major (0x00)

3. SAL_SET_VECTORS Length Argument

Replace the description for the *length_cs_n* argument for SAL_SET_VECTORS on page 102 with the following:

Bit Positions	Length in Bits	Description
0-31	32	Length of the operating system procedure in bytes (this field must be a multiple of 16).
32	1	0 = Checksum information not provided by the operating system. 1 = Checksum information provided by the operating system in bits 40-47.
33-39	7	Reserved
40-47	8	The modulo checksum of the operating system procedure code area. All bytes including the checksum byte must add up to zero.
48-63	16	Reserved.

4. SAL Procedure Return Value

Table 8-5 contains the possible return values for SAL procedures. SAL_GET_STATE_INFO may return -15 for *Retry* status. Replace Table 8-5 with the following:

Register	Conventions
0	Call completed without error.
1	Call completed without error but some information was lost due to overflow.
2	Call completed without error; effect a warm boot of the system to complete the update.
3	More information is available for retrieval.
–1	Not implemented.
–2	Invalid Argument.
–3	Call completed with error due to hardware malfunction, firmware error, or if improperly called (e.g. with PSR.cpl other than 0)
–4	Virtual address not registered.
–5	No information available.
–9	Scratch buffer required.
-15	Retry (CMC/CPE)

§